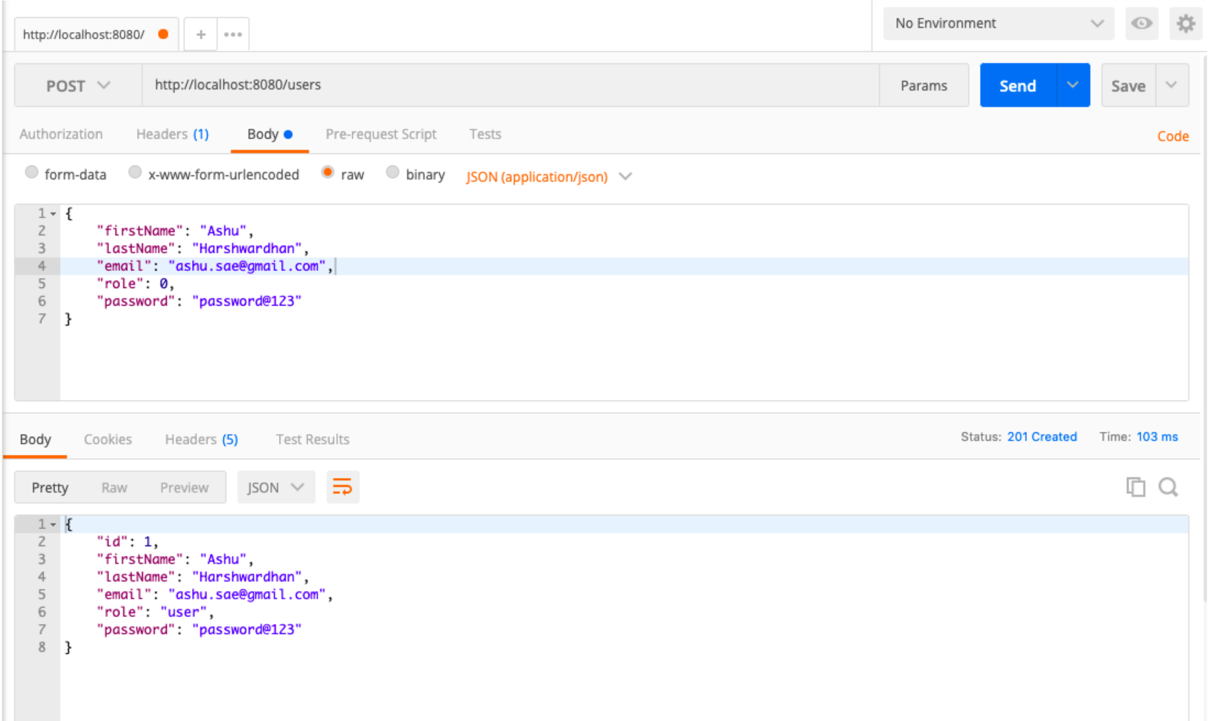
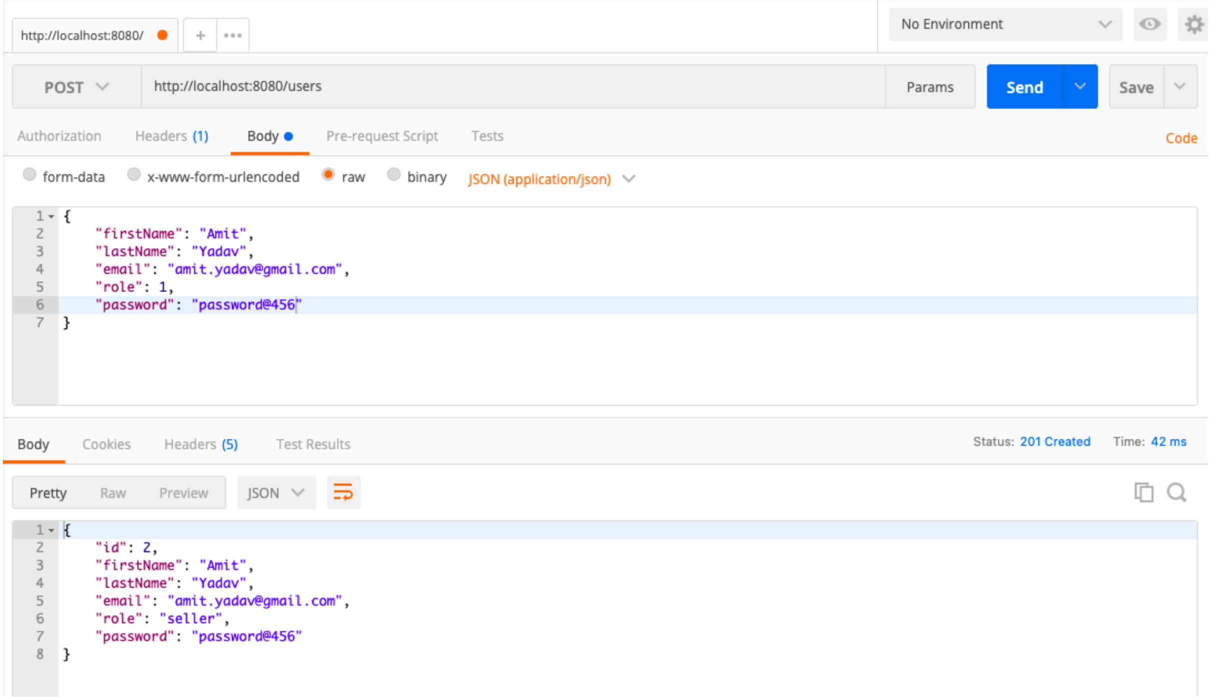
1. First we need to create users like sellers / normal users where sellers can post products i.e. create / add a new product to the application.



```
{
    "firstName": "Ashu",
    "lastName": "Harshwardhan",
    "email": "ashu.sae@gmail.com",
    "role": 0,
    "password": "password@123"
}
```

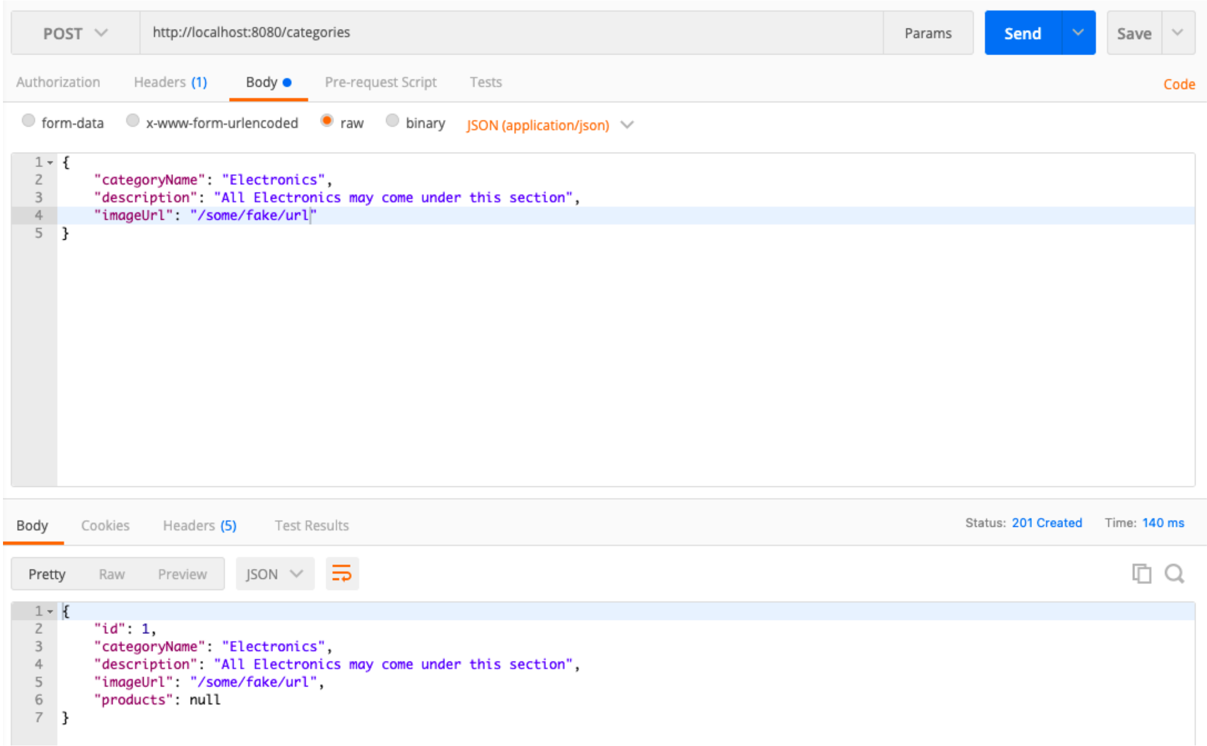
```
{
    "id": 1,
    "firstName": "Ashu",
    "lastName": "Harshwardhan",
    "email": "ashu.sae@gmail.com",
    "role": "user",
    "password": "password@123"
}
```



```
{
    "firstName": "Amit",
    "lastName": "Yadav",
    "email": "amit.yadav@gmail.com",
    "role": 1,
    "password": "password@456"
}
```

```
{
    "id": 2,
    "firstName": "Amit",
    "lastName": "Yadav",
    "email": "amit.yadav@gmail.com",
    "role": "seller",
    "password": "password@456"
}
```

2. Next, we will create a category and under the category we will add respective products using the seller account.



POST http://localhost:8080/categories

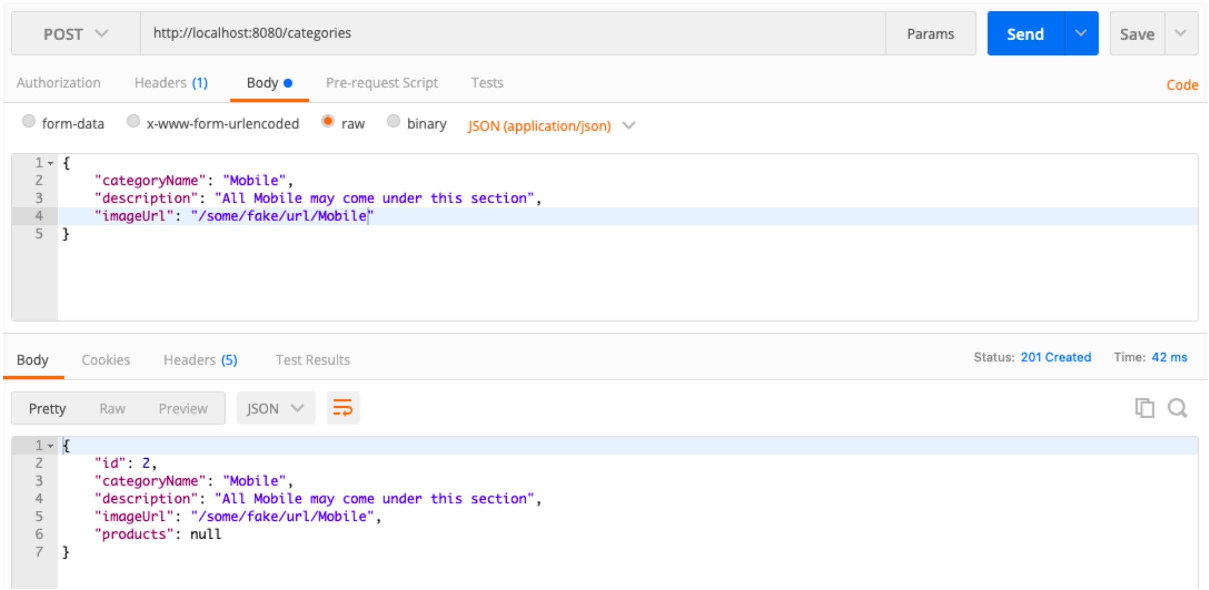
● raw JSON (application/json)

```
{
    "categoryName": "Electronics",
    "description": "All Electronics may come under this section",
    "imageUrl": "/some/fake/url"
}
```

Status: 201 Created Time: 140 ms

JSON

```
{
    "id": 1,
    "categoryName": "Electronics",
    "description": "All Electronics may come under this section",
    "imageUrl": "/some/fake/url",
    "products": null
}
```



POST http://localhost:8080/categories

● raw JSON (application/json)

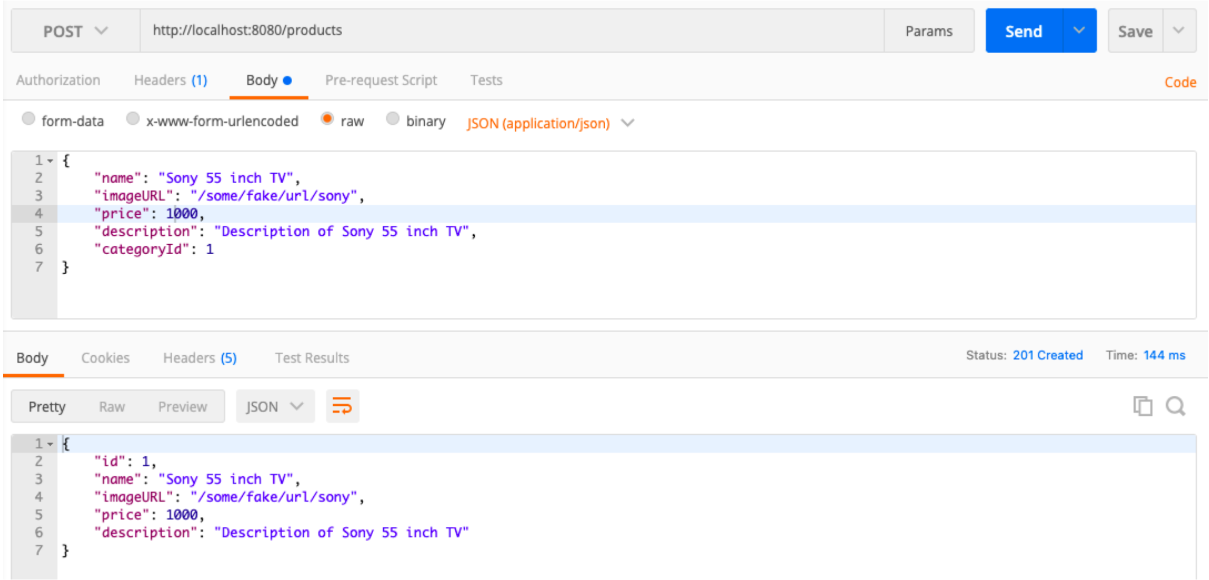
```
{
    "categoryName": "Mobile",
    "description": "All Mobile may come under this section",
    "imageUrl": "/some/fake/url/Mobile"
}
```

Status: 201 Created Time: 42 ms

JSON

```
{
    "id": 2,
    "categoryName": "Mobile",
    "description": "All Mobile may come under this section",
    "imageUrl": "/some/fake/url/Mobile",
    "products": null
}
```

3. Next, we will add products to these categories.

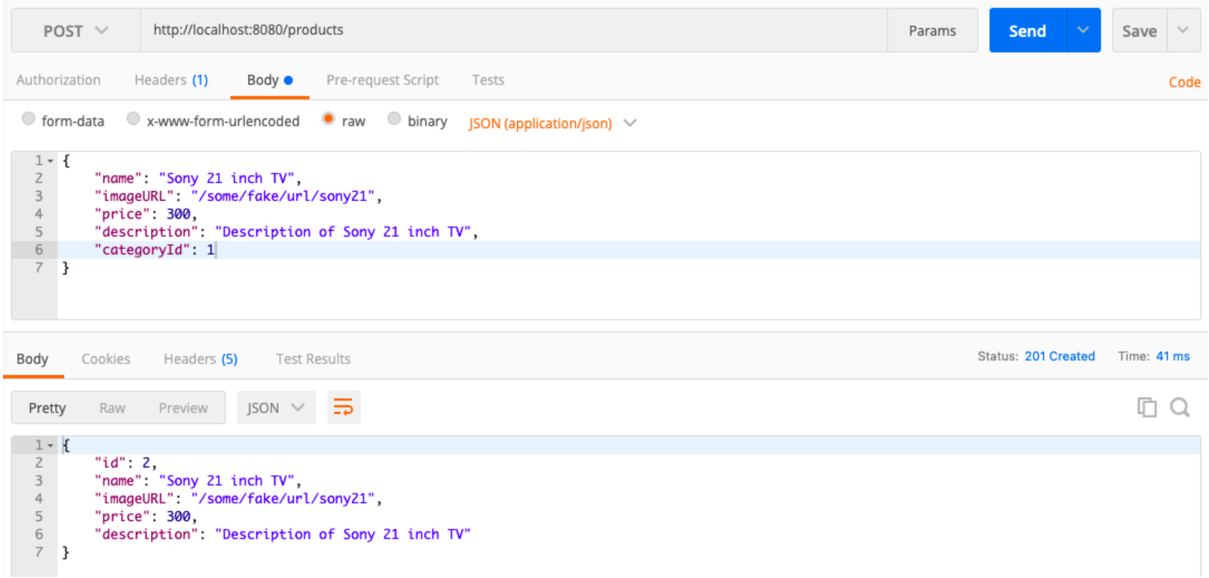


http://localhost:8080/products

```
{
    "name": "Sony 55 inch TV",
    "imageURL": "/some/fake/url/sony",
    "price": 1000,
    "description": "Description of Sony 55 inch TV",
    "categoryId": 1
}
```

Status: 201 Created Time: 144 ms

```
{
    "id": 1,
    "name": "Sony 55 inch TV",
    "imageURL": "/some/fake/url/sony",
    "price": 1000,
    "description": "Description of Sony 55 inch TV"
}
```

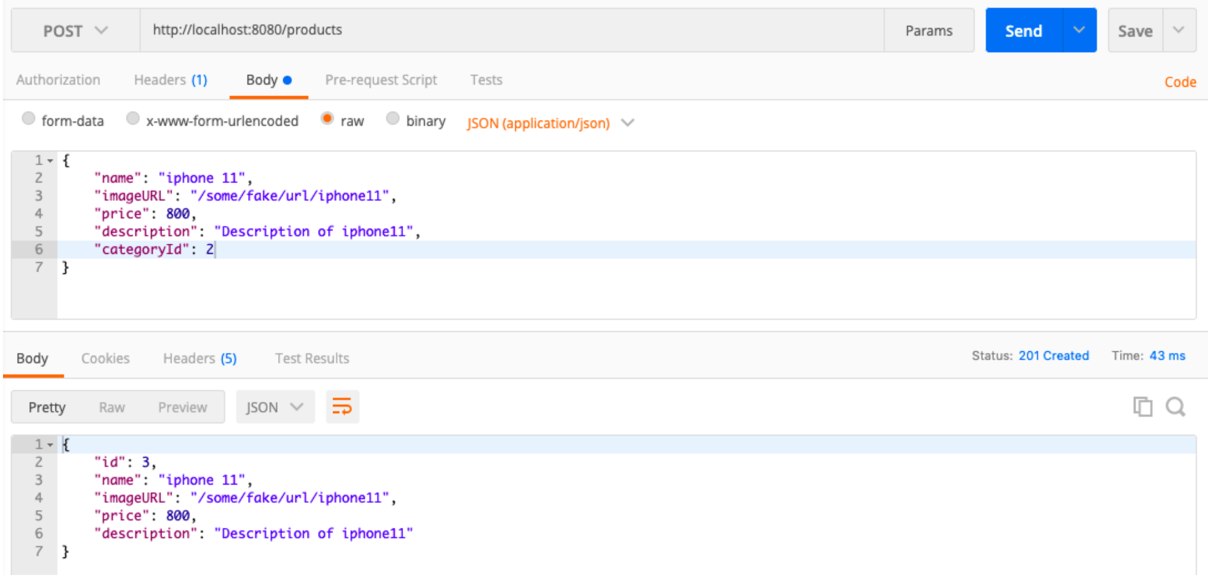


http://localhost:8080/products

```
{
    "name": "Sony 21 inch TV",
    "imageURL": "/some/fake/url/sony21",
    "price": 300,
    "description": "Description of Sony 21 inch TV",
    "categoryId": 1
}
```

Status: 201 Created Time: 41 ms

```
{
    "id": 2,
    "name": "Sony 21 inch TV",
    "imageURL": "/some/fake/url/sony21",
    "price": 300,
    "description": "Description of Sony 21 inch TV"
}
```



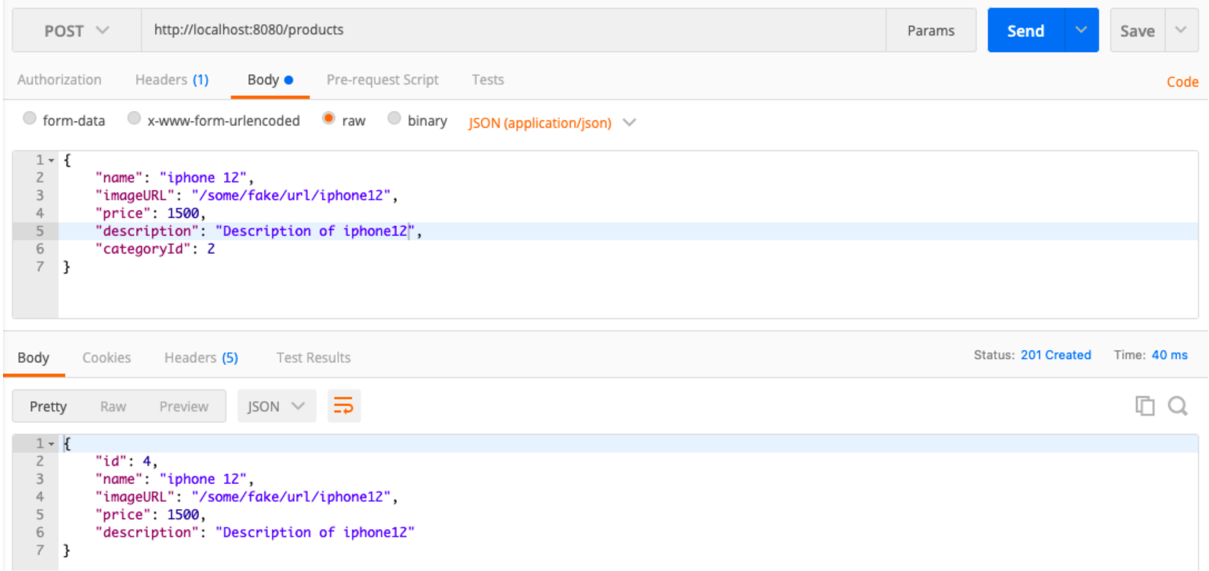
POST http://localhost:8080/products

● raw JSON (application/json)

```
{
    "name": "iphone 11",
    "imageURL": "/some/fake/url/iphone11",
    "price": 800,
    "description": "Description of iphone11",
    "categoryId": 2
}
```

Body    Status: 201 Created    Time: 43 ms

```
{
    "id": 3,
    "name": "iphone 11",
    "imageURL": "/some/fake/url/iphone11",
    "price": 800,
    "description": "Description of iphone11"
}
```



POST http://localhost:8080/products

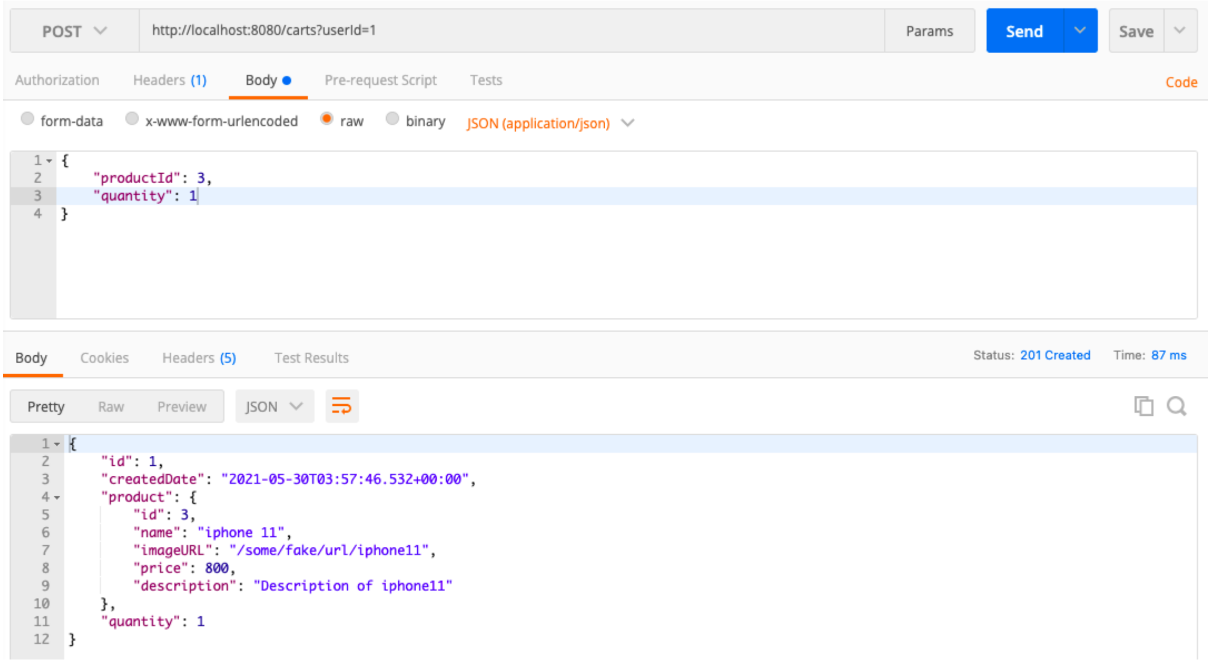
● raw JSON (application/json)

```
{
    "name": "iphone 12",
    "imageURL": "/some/fake/url/iphone12",
    "price": 1500,
    "description": "Description of iphone12",
    "categoryId": 2
}
```

Body    Status: 201 Created    Time: 40 ms

```
{
    "id": 4,
    "name": "iphone 12",
    "imageURL": "/some/fake/url/iphone12",
    "price": 1500,
    "description": "Description of iphone12"
}
```

4. Next, a normal user may add a product from a list of products and specify quantity and add that to his cart.



POST http://localhost:8080/carts?userId=1

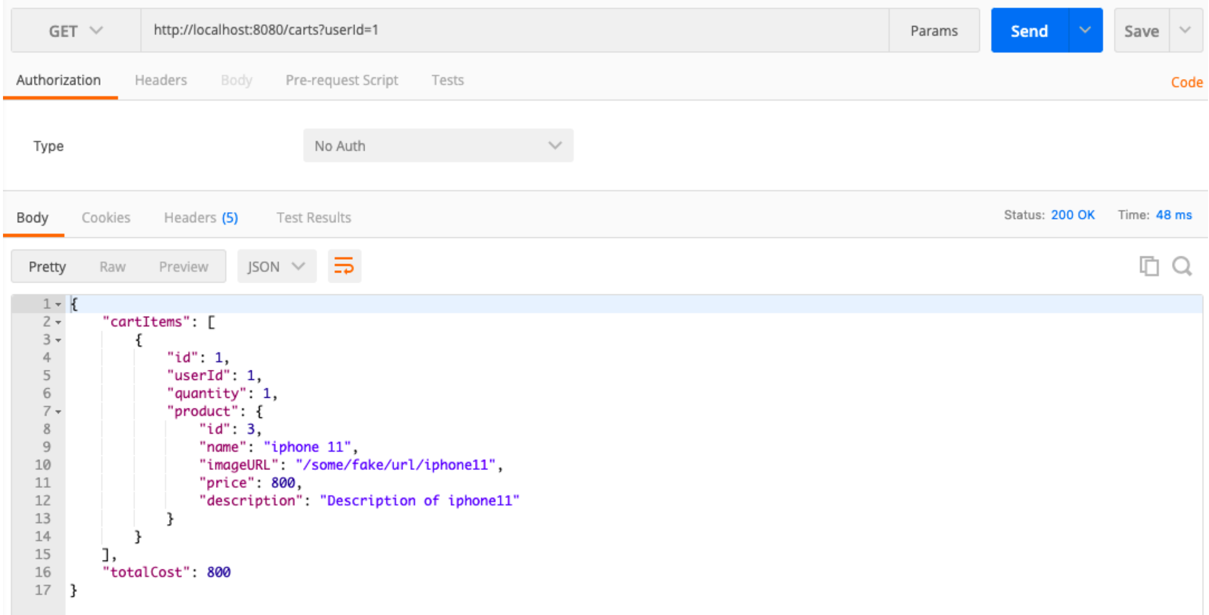
JSON (application/json)

```
{
    "productId": 3,
    "quantity": 1
}
```

Status: 201 Created Time: 87 ms

```
{
    "id": 1,
    "createdDate": "2021-05-30T03:57:46.532+00:00",
    "product": {
        "id": 3,
        "name": "iphone 11",
        "imageURL": "/some/fake/url/iphone11",
        "price": 800,
        "description": "Description of iphone11"
    },
    "quantity": 1
}
```

5. At this point of time, if the purchaser wants to view the cart and see details, we can execute a GET call to get details.



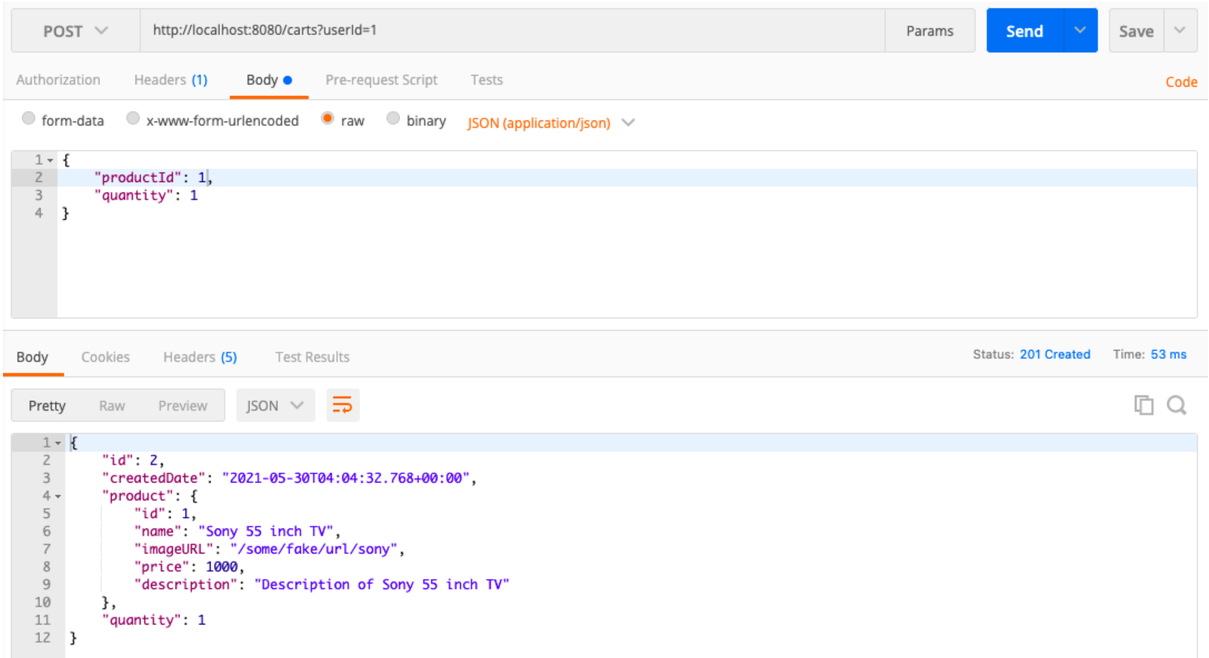
GET http://localhost:8080/carts?userId=1

Type No Auth

Status: 200 OK Time: 48 ms

```
{
    "cartItems": [
        {
            "id": 1,
            "userId": 1,
            "quantity": 1,
            "product": {
                "id": 3,
                "name": "iphone 11",
                "imageURL": "/some/fake/url/iphone11",
                "price": 800,
                "description": "Description of iphone11"
            }
        }
    ],
    "totalCost": 800
}
```

6. Let's add another product to our cart.



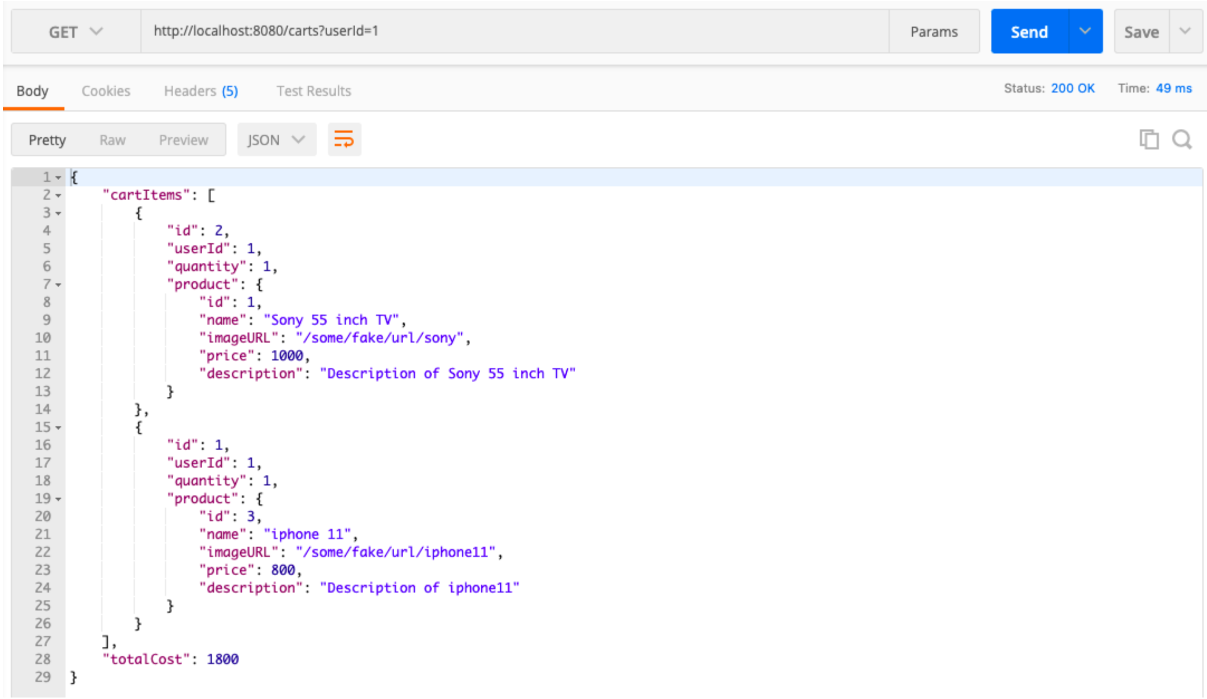
POST http://localhost:8080/carts?userId=1

```
{
    "productId": 1,
    "quantity": 1
}
```

Status: 201 Created Time: 53 ms

```
{
    "id": 2,
    "createdDate": "2021-05-30T04:04:32.768+00:00",
    "product": {
        "id": 1,
        "name": "Sony 55 inch TV",
        "imageURL": "/some/fake/url/sony",
        "price": 1000,
        "description": "Description of Sony 55 inch TV"
    },
    "quantity": 1
}
```

7. Now check cart details using the GET call again.



GET http://localhost:8080/carts?userId=1

Status: 200 OK Time: 49 ms

```
{
    "cartItems": [
        {
            "id": 2,
            "userId": 1,
            "quantity": 1,
            "product": {
                "id": 1,
                "name": "Sony 55 inch TV",
                "imageURL": "/some/fake/url/sony",
                "price": 1000,
                "description": "Description of Sony 55 inch TV"
            }
        },
        {
            "id": 1,
            "userId": 1,
            "quantity": 1,
            "product": {
                "id": 3,
                "name": "iphone 11",
                "imageURL": "/some/fake/url/iphone11",
                "price": 800,
                "description": "Description of iphone11"
            }
        }
    ],
    "totalCost": 1800
}
```

8. Finally, the purchaser can make payment and after payment, the order will be placed and the application will call POST API to place the order.